Diego Armando Sarmiento (301379640)

***Analysis report***

For the project, we are using the Olivetti faces dataset, we have loaded the data, defined the feature set, and the target column. We divided the dataset into training, testing, and validation.

A screen shot of a computer

Description automatically generated  
  
We applied PCA to reduce dimensionality.

A black rectangular object with white text

Description automatically generated  
  
We applied GaussianMixture for the covariance types full, tied, diag, and spherical. Applying the BIC method to each of these, we obtain the following values:

A screenshot of a computer program

Description automatically generated  
  
Since the type of covariance with the lowest value is Spherical, we will choose this as the best method.  
  
After, we have calculated the minimum number of clusters based on the selected type of covariance and using the BIC criterion. In this case, with 20 clusters, the lowest BIC value can be obtained, therefore this would be our most optimal number.

A black screen with a black background

Description automatically generated

*Plotting the results*

A blue line graph with white text

Description automatically generated

The variance explained curve shows that most of the variance is captured with the first 50-100 components, as the slope levels off after this range. This suggests that the dimensionality can be reduced significantly without losing much information. Using more than 100 components provides little additional value in terms of variance explained.

A bar graph with numbers and symbols

Description automatically generated

as we have seen in the previous calculations, the covariance type with the highest BIC value is full, which is considerably larger than the other types.

A graph with a line

Description automatically generated

The graph shows that the BIC consistently increases as the number of components (clusters) increases, indicating that the model with 20 components has the lowest BIC and, therefore, the best balance between fit and complexity.

when using the model to create new faces, we get a result like this:

A screenshot of a person's face

Description automatically generated

Finally, we applied modifications on a sample of the images and checked them again.

A collage of two people

Description automatically generated

Determine if the model can detect the anomalies produced in step 10 by comparing the output of the score\_samples() method for normal images and for anomalies?

A black rectangle with white text

Description automatically generated

The model can reliably detect anomalies in rotated and darkened images, as the differences in scores are noticeably negative. However, it fails to effectively detect images that are horizontally inverted, as the scores show little or no significant difference.